VSOD3.0: A Benchmark Dataset for Video Shot Occlusion Segmentation

Fengyin Huang[123], Junhua Liao[4], Feng Liang[12], Chenyang Wang[5], and Haihan Duan[12*]

[1] Artificial Intelligence Research Institute, Shenzhen MSU-BIT University, Shenzhen, China
[2] Guangdong-Hong Kong-Macao Joint Laboratory for Emotion Intelligence and Pervasive Computing, Shenzhen, China
[3] School of Automation, Beijing Institute of Technology, Beijing, China
[4] School of Computer Science, Sichuan University, Chengdu, China
[5] School of Computer and Electronic Information / School of Artificial Intelligence, Nanjing Normal University, Nanjing, China
bravehfy@163.com, 495750571@qq.com, fliang@smbu.edu.cn, chenyangwang@ieee.org, duanhaihan@smbu.edu.cn
*Corresponding author: Haihan Duan (e-mail: duanhaihan@smbu.edu.cn)

Abstract. As the demand for automated video editing continues to rise, accurately identifying occlusions within video segments has become a significant challenge. However, existing video occlusion datasets have notable drawbacks, such as an excessive proportion of occlusions in video clips, overly short segment durations, and inaccurate occlusion annotations. More importantly, current video occlusion detection datasets can only determine whether frames are occluded, rather than pinpointing the exact location of the occlusions. To address these limitations, we have developed VSOD3, a new benchmark dataset for video shot occlusion segmentation, which features pixel-level precise annotations. Additionally, we propose a video occlusion segmentation method to validate this dataset. Extensive experiments conducted on our dataset, as well as other existing datasets, demonstrate that our method outperforms state-of-the-art approaches across multiple relevant metrics.

Keywords: video shot occlusion segmentation · video semantic segmentation · target-guided temporal fusion.

1 Introduction

Accurately localizing occluded frames and occluded regions in videos can facilitate efficient video post-processing and improve the usability of video content in practical applications. However, video shot occlusion segmentation is far from trivial. In video scenes, the boundaries of occluded regions are usually fragmented and ambiguous, making them difficult to capture with conventional segmentation methods. In addition, occlusions in videos evolve continuously over time, which makes single-frame prediction prone to temporal fluctuation.

In recent years, many studies have investigated video occlusion detection [7–9]. Most of these works mainly focus on identifying which frames in a video clip contain occlusion. Although some of them can detect nearly all occluded frames with high accuracy, they are unable to further localize the exact occluded regions within those frames. On the one hand, this limitation is related to the annotation granularity of existing public video occlusion datasets, which usually do not provide pixel-level labels for occluded regions. The earliest video occlusion detection dataset, VSOD1[7] (*Video Shot Occlusion Detection v1.0*), provides bounding-box annotations for occluded regions. However, occlusion occupies a large temporal proportion in these videos, meaning that most frames in the dataset contain occlusions, which does not well reflect real-world video scenarios. Moreover, the authors of VSOD1 did not use the annotated occlusion regions in [7, 8]; instead, they simply formulated the task as a frame-level classification problem, where each frame is categorized as either occluded or non-occluded. The authors of VSOD1 later introduced VSOD2[9] (*Video Shot Occlusion Detection v2.0*), which is also designed for video occlusion detection. Unlike VSOD1, VSOD2 reduces the temporal proportion of occlusions and is therefore more consistent with practical video editing requirements. However, it still does not address the localization of specific occluded regions, since the task remains essentially a frame-level classification problem.

On the other hand, these limitations that these methods cannot localize occluded regions suggest that detection-based formulations are no longer sufficient for the current demand of fine-grained occlusion analysis. Instead, semantic segmentation provides a more appropriate paradigm for locating occluded regions at the pixel level. Nevertheless, existing video semantic segmentation methods [15, 16, 3] are mostly designed for general multi-class segmentation and are not well suited to occlusion segmentation, which is characterized by sparse foregrounds, boundary sensitivity, and temporally unstable appearance changes.

To address these issues, we construct a new dataset dedicated to video shot occlusion segmentation, named **VSOD3** (*Video Shot Occlusion Detection v3.0*). VSOD3 inherits the advantages of both VSOD1 and VSOD2: it maintains a more reasonable temporal proportion of occlusions while providing pixel-level annotations for occluded regions. Therefore, it enables the training and evaluation of semantic segmentation methods for video shot occlusion analysis.Based on a multi-scale feature extraction backbone, we further propose an occlusion segmentation method that is more suitable for binary segmentation scenarios. Specifically, we introduce a target-guided temporal fusion mechanism, an explicit boundary enhancement branch, and a joint loss optimization strategy to improve occlusion discrimination, boundary recovery, and temporal stability.

In summary, the main contributions of this work are as follows:

- We construct VSOD3, a dedicated benchmark dataset for video shot occlusion segmentation. VSOD3 provides frame-by-frame pixel-level annotations of occluded regions, enabling systematic training and evaluation.
- We propose a segmentation architecture tailored to video shot occlusion segmentation. The architecture consists of three key modules: a multi-scale

feature decoding module, a target-guided temporal fusion module, and a boundary enhancement module, which jointly improve occlusion localization.
– Extensive experiments are conducted on the proposed VSOD3 benchmark. The results demonstrate that our method achieves superior performance compared with representative baselines, especially in suppressing false positives and improving occluded region segmentation quality.

2 Related Work

2.1 Video Occlusion Detection

Occlusion has been widely studied in different computer vision tasks. Some works focus on *the detection of occluded objects* [1, 14, 17, 21], where the goal is to detect target objects even when they are partially occluded. Other studies investigate *occluded person re-identification*[4, 19], which aims to match pedestrian identities under partial occlusion. There are also works on occlusion modeling and instance segmentation[6], where the objective is to explicitly reason about occlusion relationships among instances. Although these tasks are closely related to occlusion, their main focus is still the occluded target object itself.

In contrast, video occlusion detection aims to identify *occlusion events* in videos rather than specific target objects. Therefore, it relies more heavily on the effective utilization of temporal information across consecutive frames. Liao *et al.*[7, 8] explored deep-learning-based video occlusion detection methods by extracting spatio-temporal features from videos, and achieved high detection accuracy with real-time processing speed on public datasets. In addition, they proposed a non-data-driven occlusion detection algorithm based on fluctuations of depth information[9], motivated by the observation that occlusion often occurs in the foreground and is therefore accompanied by depth variation.

These video occlusion detection methods have achieved promising results, however,they are mainly designed to determine whether an occlusion event occurs in a given frame. They can localize the frames where occlusion happens, but they cannot further identify the exact occluded regions within those frames. To achieve such fine-grained localization, it is necessary to introduce segmentation-based methods, especially video semantic segmentation techniques.

2.2 Video Semantic Segmentation

Video semantic segmentation mainly focuses on how to effectively exploit temporal information across consecutive frames. Common strategies include optical-flow-based propagation[12], memory modules[13], spatio-temporal Transformers[20, 10], temporal attention[16, 15], and more recently, state-space models[3].

To better utilize spatio-temporal contextual information in videos, Sun *et al.*[15] proposed the CFFM (*Coarse-to-Fine Feature Mining*)method, which jointly models static scene information and motion information through cross-frame

feature mining. Different from methods that directly establish frame correspondence through optical flow or attention mechanisms, MRCFA[16](*Mining Relations among Cross-Frame Affinities*) further investigates higher-order cross-frame relations and enhances temporal feature propagation through relation refinement and multi-scale aggregation. TV3S[3](*Temporal Video State Space Sharing*), proposed by Hesham *et al.*, employs the Mamba state-space model to build a long-range temporal feature sharing mechanism, enabling efficient spatio-temporal modeling without requiring an additional memory bank, and thus improving temporal consistency and segmentation accuracy in long videos.

Although these methods are effective in enhancing temporal representation and exploiting frame continuity, they are primarily designed for general multi-class video semantic segmentation. For binary video shot occlusion segmentation, temporal modeling complexity is not the only concern. Instead, foreground discrimination, boundary recovery, and prediction stability are particularly important. Motivated by this observation, we propose a target-guided temporal fusion framework for video shot occlusion segmentation, which is more closely aligned with the characteristics of binary occlusion scenes than heavy generic temporal modeling methods such as TV3S[3].

3 Method

3.1 Overview

In this section, we introduce the proposed framework for video shot occlusion segmentation, which consists of a shared backbone, a multi-scale projection and fusion module, a target-guided temporal fusion module, a boundary enhancement branch, and a final segmentation head. The overall architecture is illustrated in Fig. 1. Given an input video clip, the framework first extracts and fuses multi-scale feature by SegFormer-B1 backbone[18], and then enhances the fused representation through the temporal fusion module and the boundary enhancement branch. Finally, the model predicts the occlusion segmentation result of the target frame and is optimized with a joint loss during training.

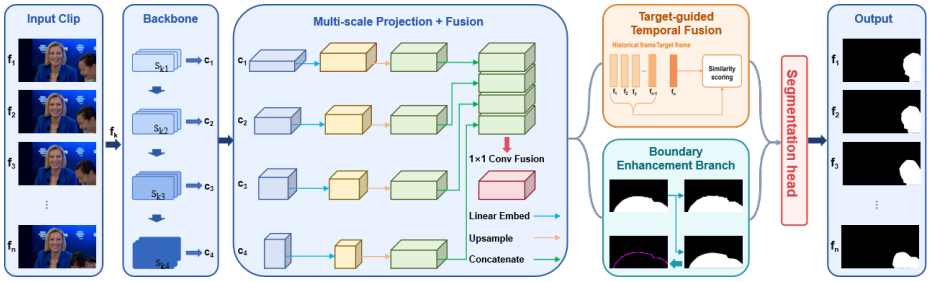

Fig. 1. Overview of our proposed network framework

3.2 Target-guided Temporal Fusion

Unlike many other segmentation tasks, video shot occlusion segmentation exhibits strong temporal characteristics, as the location of occlusion regions often vary over time. Therefore, historical frame information is important for predicting the current frame. However, not all historical frames contribute equally to the target frame, which calls for a target-centered selective fusion mechanism.

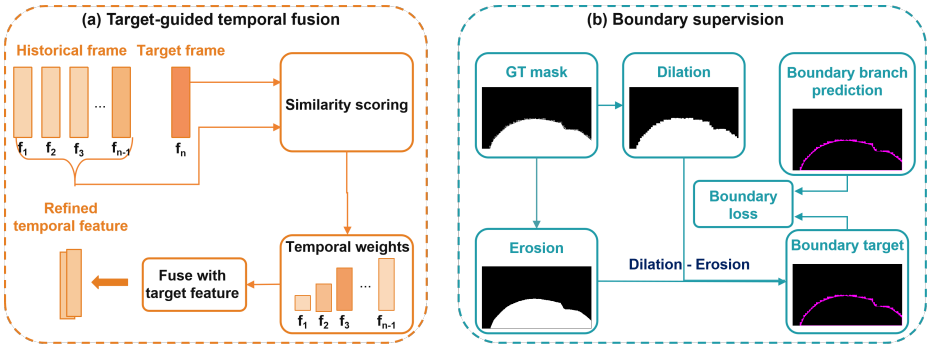

Fig. 2. Details of temporal fusion and boundary supervision

To address this issue, we propose the *Target-guided Temporal Fusion* module, as shown in Fig. 2 (a). After the input video clip is processed by the backbone and the multi-scale projection and fusion module, we obtain the fused clip features. The last frame of the clip is treated as the target feature, while the preceding frames are considered historical features. Each historical feature is concatenated and differenced from the target feature, and the resulting feature is fed into a scoring module to estimate the importance weight of each frame.

The weight assignment is mainly determined by the relevance between each historical frame and the target feature, where frames that are more consistent with the target frame receive larger weights. This design is motivated by the temporal characteristics of video occlusion scenes: adjacent frames usually exhibit strong temporal continuity, while occlusion events often correspond to local appearance changes. In most cases, the background and non-occluded regions remain relatively stable within a short temporal window. Therefore, historical frames that are more similar to the target frame tend to provide more reliable contextual cues for accurately identifying and localizing occluded regions. The formula for determining the weights is as follows:

$$s_t = \phi([\mathbf{F}_t; \mathbf{F}_T; \mathbf{F}_t - \mathbf{F}_T]), \qquad w_t = \frac{\exp(s_t)}{\sum_{k=1}^{T} \exp(s_k)}, \tag{1}$$

where $\mathbf{F}_t \in \mathbb{R}^{C\times H\times W}$ denotes the feature map of the t-th historical frame, $\mathbf{F}_T \in \mathbb{R}^{C\times H\times W}$ denotes the feature map of the target frame, and T is the total

number of frames in the input clip. The operator $[\cdot\,;\cdot]$ denotes channel-wise concatenation, and $\mathbf{F}_t - \mathbf{F}_T$ represents the feature difference between the historical frame and the target frame. $\phi(\cdot)$ denotes the scoring function implemented by the temporal attention module, s_t is the frame-wise relevance score, and w_t is the normalized temporal weight obtained by applying the softmax function.

After the frame-wise scores are estimated, they are normalized along the temporal dimension and used to compute a weighted sum of the clip features. The resulting temporal context is then fused with the target-frame feature to obtain the refined temporal feature. The process is formulated as follows:

$$\mathbf{F}_{\text{ctx}} = \sum_{t=1}^{T} w_t \odot \mathbf{F}_t, \qquad \mathbf{F}_{\text{ref}} = \psi([\mathbf{F}_T; \mathbf{F}_{\text{ctx}}]), \tag{2}$$

where $\mathbf{F}_{\text{ctx}}$ denotes the aggregated temporal context feature, $\odot$ denotes element-wise weighting between the temporal weight and the corresponding frame feature, and $\mathbf{F}_{\text{ref}}$ denotes the refined temporal feature used for subsequent segmentation. $\psi(\cdot)$ represents the fusion operation that combines the target-frame feature $\mathbf{F}_T$ and the aggregated temporal context $\mathbf{F}_{\text{ctx}}$.

Compared with heavy generic temporal modeling methods[3], the proposed Target-guided Temporal Fusion module focuses more directly on historical information that is relevant to the current frame. This design reduces the interference of irrelevant frames and improves the prediction stability for the target frame.

3.3 Boundary Enhancement

By examining the dataset, we observe that occlusion regions are often thin, ambiguous, and irregular. Relying solely on the main segmentation branch may therefore lead to blurred boundaries or merged regions. To alleviate this problem, we introduce the *Boundary Enhancement* branch, as illustrated in Fig. 2 (b).

Specifically, a boundary target is generated from the ground-truth mask through morphological operations. We first apply dilation and erosion to the binary mask and then compute their difference to get the boundary region:

$$\mathbf{B}_{\text{gt}} = \delta(\mathbf{M}_{\text{gt}}) - \varepsilon(\mathbf{M}_{\text{gt}}), \tag{3}$$

where $\mathbf{M}_{\text{gt}} \in \{0,1\}^{H\times W}$ denotes the binary ground-truth occlusion mask, $\delta(\cdot)$ denotes the dilation operation, $\varepsilon(\cdot)$ denotes the erosion operation, and $\mathbf{B}_{\text{gt}}$ denotes the generated boundary target.

Based on the boundary-sensitive feature, the boundary branch predicts a boundary logit, which is then used to compute the boundary loss:

$$\mathcal{L}_{\text{boundary}} = \frac{1}{N}\sum_{i=1}^{N} \text{BCE}\Big(\mathbf{B}_{\text{pred}}^{(i)}, \mathbf{B}_{\text{gt}}^{(i)}\Big), \tag{4}$$

where $\mathbf{B}_{\text{pred}}$ denotes the predicted boundary logit, $\mathbf{B}_{\text{gt}}$ denotes the ground-truth boundary target, $\text{BCE}(\cdot,\cdot)$ denotes the binary cross-entropy loss, and N is the number of valid pixels used for normalization.

The learned boundary information is further injected into the main segmentation branch to refine the final segmentation result. In this way, the Boundary Enhancement module explicitly strengthens the learning of contour regions, thereby improving the quality of small occlusion regions and boundary recovery.

3.4 Joint Loss Function

The overall training objective consists of three components: the cross-entropy loss, the Dice loss, and the boundary loss. The total loss is defined as

$$\mathcal{L} = \mathcal{L}_{\text{ce}} + \lambda_{\text{dice}}\mathcal{L}_{\text{dice}} + \lambda_{\text{boundary}}\mathcal{L}_{\text{boundary}}, \tag{5}$$

where $\mathcal{L}_{\text{ce}}$ denotes the cross-entropy loss, $\mathcal{L}_{\text{dice}}$ denotes the Dice loss, and $\mathcal{L}_{\text{boundary}}$ denotes the boundary loss. λ_{dice} and $\lambda_{\text{boundary}}$ are the weighting coefficients for the Dice loss and boundary loss, respectively.

Boundary Loss The boundary loss supervises the boundary branch, strengthens boundary-aware learning, and improves the fineness of contour prediction.The detailed calculation of the boundary loss is given in Eq. 4.

Cross-Entropy Loss The cross-entropy loss serves as the basic supervision for pixel-wise classification and ensures the overall stability of segmentation learning. It is defined as

$$\mathcal{L}_{\text{ce}} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c}\log p_{i,c}, \tag{6}$$

where N denotes the number of valid pixels, C denotes the number of classes, $y_{i,c}$ is the one-hot ground-truth label of pixel i for class c, and $p_{i,c}$ is the predicted probability for class c at pixel i. In our task, $C = 2$, corresponding to the non-occlusion and occlusion classes.

Dice Loss The Dice loss[11] is introduced to alleviate foreground sparsity and class imbalance. It focuses more on the overlap between the predicted occlusion region and the ground-truth region, which helps improve the recall of occlusion foregrounds. It is formulated as

$$\mathcal{L}_{\text{dice}} = 1 - \frac{2\sum_{i=1}^{N}\hat{y}_i y_i + \epsilon}{\sum_{i=1}^{N}\hat{y}_i + \sum_{i=1}^{N} y_i + \epsilon}, \tag{7}$$

where $\hat{y}_i$ denotes the predicted probability of the occlusion class at pixel i, $y_i \in \{0, 1\}$ denotes the corresponding binary ground-truth label, N is the number of valid pixels, and ϵ is a small constant for numerical stability.

Overall, the three loss terms play complementary roles in optimization: the cross-entropy loss ensures basic segmentation stability, the Dice loss improves foreground recall, and the boundary loss refines contour quality. Their combination is therefore more suitable for binary video shot occlusion segmentation.

4 Dataset

Video shot occlusion has attracted increasing attention in recent years, and several related datasets have been introduced, such as VSOD1[7] and VSOD2[9]. However, these datasets still have limitations for studying video shot occlusion segmentation. In VSOD1, occlusion occupies a relatively large portion of the videos, making the scenarios less balanced. In VSOD2, individual videos are relatively long, which increases the difficulty of training and evaluation. More importantly, both datasets mainly focus on whether occlusion occurs, rather than providing precise pixel-level localization of the occluded regions.

To address these limitations, we construct a new **pixel-level video shot occlusion segmentation dataset**, namely **VSOD3** (*Video Shot Occlusion Dectection v3.0*). VSOD3 is designed to support both temporal localization of occluded clips and spatial localization of occluded regions within frames, and therefore serves as a benchmark for evaluating video shot occlusion segmentation methods. Representative examples and their annotations are shown in Fig. 3.

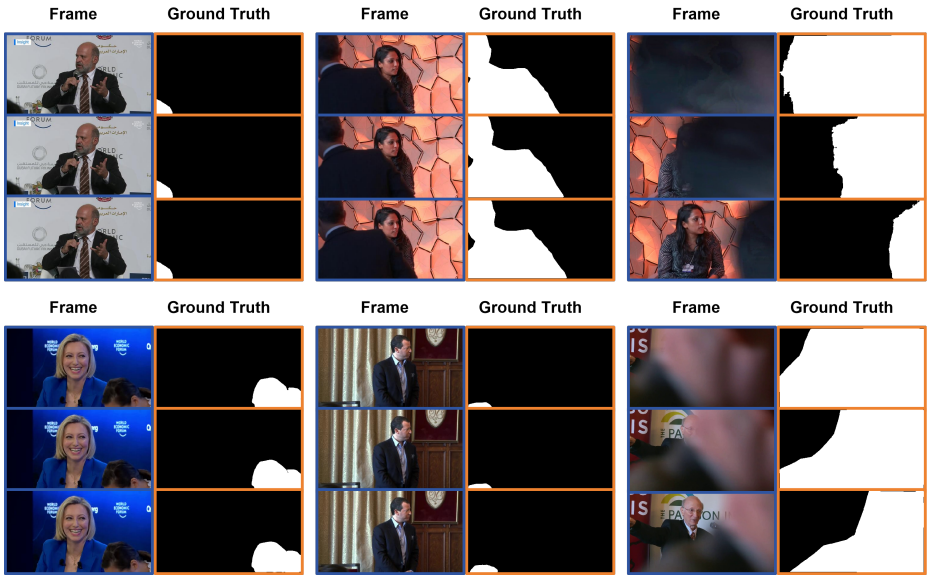

Fig. 3. Examples of original frames and corresponding annotations in the VSOD3

4.1 Dataset Annotation

VSOD3 contains 2,048 video clips, among which 1,630 clips are used for training and 408 clips are reserved for testing. To precisely localize the occluded regions,

all annotations were manually created by several annotators with a computer vision background over a period of several months.

The final annotations are provided as pixel-wise binary mask images in PNG format. In each mask, a pixel value of 1 denotes an occluded pixel, while a pixel value of 0 denotes a non-occluded pixel. Such dense pixel-level annotations enable segmentation-level evaluation of video shot occlusion, which distinguishes VSOD3 from previous datasets mainly oriented toward occlusion detection.

4.2 Dataset Statistics

To better simulate realistic video shot occlusion scenarios, we collected diverse samples covering a wide range of common occlusion events in real videos, including pedestrians passing in front of the camera, unexpected objects entering the screen, and cameras or foreground objects blocking the main speaker.

VSOD3 contains more than 2,000 video clips, with an average duration of 8.664 seconds and an average length of 216.6 frames per clip. On average, 16.6% of frames in each clip contain occlusion. Moreover, among the occluded frames, the average occlusion area percentage is 28.0%, indicating that 28.0% of pixels are labeled as occluded on average in frames where occlusion occurs. Fig. 4 summarizes the statistical characteristics of VSOD3 from the perspectives of video length, percentage of occluded frames, and occlusion area percentage.

Compared with previous datasets, VSOD3 not only provides finer pixel-level annotations, but also contains more realistic and diverse occlusion patterns. Overall, it provides a realistic, diverse, and densely annotated benchmark for evaluating performance in video shot occlusion segmentation.

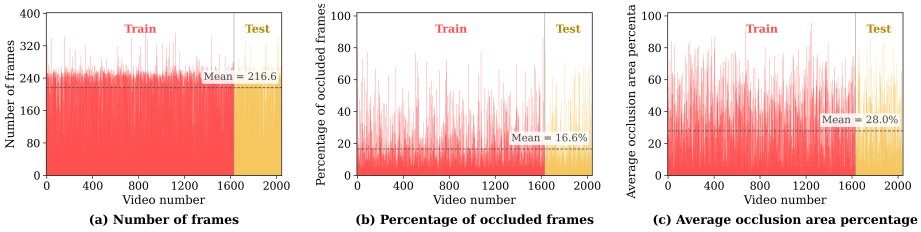

Fig. 4. Statistics of VSOD3

5 Experiments

5.1 Implementation Details

All experiments are conducted on a remote server with an **NVIDIA GeForce RTX 4090 GPU (24 GB memory)** under an **Ubuntu 24.04** virtual machine environment. The CUDA version is **12.9**. For optimization, we use the

AdamW optimizer with an initial learning rate of 6×10^{-5}, momentum parameters `betas` $= (0.9, 0.999)$, and a weight decay of **0.01**. The whole training process is conducted for a total of **160000 iterations**.

5.2 Parameter Sensitivity Analysis

Table 1 reports the sensitivity of our method to the Dice weight d_w and boundary weight b_w. To clearly reveal the influence of each coefficient, we present two representative parameter sweeps: varying b_w while fixing $d_w = 0.8$, and varying d_w while fixing $b_w = 0.3$. When $d_w = 0.8$, increasing b_w from 0.1 to 0.3 consistently improves the segmentation quality, with IoU_s and $\mathrm{F1}_s$ reaching 0.9014 and 0.9259, respectively. A larger boundary weight does not bring further improvement and instead reduces the foreground-level performance, indicating that excessive boundary supervision may disturb the region prediction.

When fixing $b_w = 0.3$, the results show that both insufficient and excessive Dice weight degrade the performance. Although $d_w = 0.6$ achieves slightly higher foreground-level scores, it also produces a higher false positive rate. In comparison, $d_w = 0.8$ provides a more balanced trade-off between accuracy and false alarm suppression, achieving high IoU_s and $\mathrm{F1}_s$ while maintaining a low FPR. Based on this trade-off, we finally select $d_w = 0.8$ and $b_w = 0.3$ for further training. With the selected setting, the final model further improves IoU_s from 0.9014 to 0.9158 and IoU_f from 0.8018 to 0.8436, while keeping the false positive rate at a low level. These results demonstrate that the selected parameter combination offers stable and effective supervision for the final training stage.

Table 1. Parameter sensitivity analysis of different Dice weight (d_w) and boundary weight (b_w). The selected setting is marked in gray.

d_w	b_w	$\mathrm{IoU}_s \uparrow$	$\mathrm{F1}_s \uparrow$	$\mathrm{FPR}_s \downarrow$	$\mathrm{IoU}_f \uparrow$	$\mathrm{F1}_f \uparrow$	$\mathrm{FPR}_f \downarrow$
Varying b_w with $d_w = 0.8$							
0.8	0.1	0.7979	0.8542	0.1155	0.7960	0.8614	0.0424
0.8	0.2	0.8872	0.9184	0.0355	0.8425	**0.8967**	0.0128
0.8	0.3	**0.9014**	**0.9259**	0.0174	0.8018	0.8676	0.0062
0.8	0.4	0.8763	0.9080	0.0219	0.7419	0.8232	0.0136
0.8	0.5	0.8900	0.9184	**0.0134**	0.7784	0.8510	**0.0040**
Varying d_w with $b_w = 0.3$							
0.5	0.3	0.8606	0.8911	**0.0067**	0.6744	0.7630	**0.0020**
0.6	0.3	**0.9087**	**0.9331**	0.0268	**0.8380**	**0.8924**	0.0124
0.7	0.3	0.8248	0.8663	0.0102	0.6207	0.7289	0.0040
0.8	0.3	0.9014	0.9259	0.0174	0.8018	0.8676	0.0062
0.9	0.3	0.8094	0.8497	0.0069	0.6389	0.7321	0.0020
Further training with the selected setting							
0.8	0.3	**0.9158**	**0.9361**	0.0212	**0.8436**	**0.8932**	0.0089

5.3 Comparison with State-of-the-Art Methods

Tables 2 and 3 report the comparison results with state-of-the-art methods. To comprehensively evaluate the model, we both adopt sequence-level metrics and frame-level metrics. The compared methods include classical convolutional networks (ResNet-101[2] and DenseNet-169[5]), representative video occlusion detection methods (SAT module[8] and Liao *et al.*[9]), and representative video semantic segmentation methods (VSS-CFFM[15], TV3S[3], and MRCFA[16]).

From Tables 2 and 3, the proposed method shows clear advantages on several key dimensions of video shot occlusion segmentation. In particular, our method achieves the lowest false positive rates at both the sequence level and the frame level, with FPR_s of 0.0212 and FPR_f of 0.0089, respectively. This indicates that the proposed method can more effectively distinguish true occlusion regions from background regions, thereby significantly reducing false positives.

Meanwhile, the proposed method achieves the best sequence-level IoU, Accuracy, and F1-score, reaching 0.9158, 0.9659, and 0.9361, respectively, which demonstrates stronger temporal stability and overall consistency across consecutive frames. At the frame level, our method also obtains the best $\mathrm{Accuracy}_f$

Table 2. Comparison with state-of-the-art methods on the VSOD3 dataset in terms of sequence-level metrics. The best results are highlighted in bold.

Method	Recall_s	IoU_s	$\mathrm{Accuracy}_s$	FPR_s	Recall_s^E	$\mathrm{Precision}_s$	$\mathrm{F1}_s$
ResNet-101[2]	0.9948	0.6756	0.8775	0.1484	0.9479	0.7016	0.8064
DenseNet-169[5]	0.9974	0.6514	0.8638	0.1665	0.9458	0.6766	0.7889
SAT module[8]	**1.0000**	0.4762	0.7129	0.3817	0.9698	0.4834	0.6452
Liao et al .	0.9314	0.2587	0.3109	0.9209	0.9204	0.2605	0.3835
VSS-CFFM[15]	0.9828	0.7988	0.9019	0.1105	0.9417	0.8323	0.8534
TV3S (mit-b1)[3]	0.9877	0.7419	0.8310	0.2165	0.9607	0.7673	0.8052
TV3S (swin-s)[3]	0.9902	0.8214	0.9136	0.1108	**0.9798**	0.8308	0.8734
MRCFA (mit-b2)[16]	0.9902	0.8772	0.9323	0.0876	0.9713	0.8949	0.9105
Ours	0.9828	**0.9158**	**0.9659**	**0.0212**	0.9362	**0.9592**	**0.9361**

Table 3. Comparison with state-of-the-art methods on the VSOD3 dataset in terms of frame-level metrics. The best results are highlighted in bold.

Method	Recall_f^E	Recall_f	IoU_f	$\mathrm{Accuracy}_f$	FPR_f	$\mathrm{Precision}_f$	$\mathrm{F1}_f$
ResNet-101[2]	0.9784	0.8657	0.6205	0.9166	0.0738	0.6866	0.7658
DenseNet-169[5]	0.9760	0.8633	0.5584	0.8925	0.1020	0.6125	0.7166
SAT module[8]	**0.9856**	0.9191	0.5181	0.8654	0.1446	0.5428	0.6825
Liao et al .[9]	0.9191	0.4481	0.1277	0.5346	0.4496	0.1650	0.2148
VSS-CFFM[15]	0.9739	0.8612	0.7685	0.9411	0.0397	0.8769	0.8427
TV3S (mit-b1)[3]	0.9800	0.9132	0.7380	0.8748	0.1359	0.8031	0.8060
TV3S (swin-s)[3]	0.9833	**0.9667**	**0.8708**	0.9632	0.0387	0.8916	**0.9123**
MRCFA (mit-b2)[16]	0.9833	0.9351	0.8662	0.9518	0.0477	0.9181	0.9082
Ours	0.9698	0.8591	0.8436	**0.9662**	**0.0089**	**0.9623**	0.8932

(0.9662) and Precision$_f$ (0.9623), while maintaining strong IoU$_f$ and F1$_f$, indicating that it can localize occluded regions more reliably and produce more precise pixel-level predictions. Although the proposed method is slightly lower than some more aggressive methods on several recall-related metrics, it achieves a better balance among false positive suppression, temporal stability, and localization reliability, making it better suited to the practical requirements.

Overall, these results demonstrate that the proposed target-guided temporal fusion and boundary enhancement design is more suitable for video shot occlusion segmentation than directly adopting generic video segmentation methods.

5.4 Qualitative Results

Fig. 5 compares the input frames, ground-truth annotations, and the predictions of different video semantic segmentation methods.The red regions denote the predicted occlusion masks. As can be observed, several competing methods tend to activate large non-occluded areas, especially around object boundaries or visually salient foreground regions. Such over-detection indicates that these methods are easily affected by appearance changes, motion blur, and foreground-background ambiguity, resulting in a relatively high false positive rate.

In contrast, our method produces more conservative and precise occlusion predictions. The detected regions are mainly concentrated on truly occluded areas, while most visible object and background regions are correctly suppressed. This demonstrates that our approach can better distinguish real occlusions from normal foreground structures, reducing spurious responses in non-occluded regions. Moreover, the predictions of our method remain spatially coherent across different frames, suggesting improved temporal robustness under dynamic scenes.

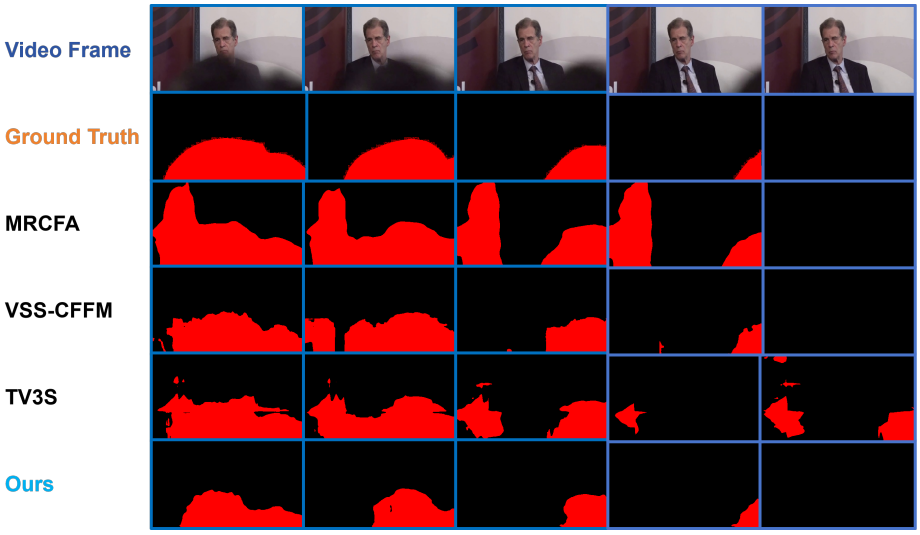

Fig. 5. Qualitative Results

Overall, the qualitative comparison confirms the effectiveness of our method in suppressing false alarms while preserving meaningful occlusion regions. This advantage is particularly important for various downstream video understanding tasks, where excessive false positives may introduce noisy cues.

6 Conclusion

In this work, we addressed the task of video shot occlusion segmentation by proposing a dedicated segmentation framework tailored to binary occlusion scenarios. Built upon a multi-scale feature extraction backbone, the proposed method incorporates a target-guided temporal fusion module to selectively aggregate useful historical information, a boundary enhancement branch to improve contour recovery, and a joint loss to alleviate foreground sparsity and boundary ambiguity. In addition, we constructed the VSOD3 dataset to support fine-grained evaluation of occlusion segmentation at both the temporal and spatial levels.

Experimental results on VSOD3 demonstrate the effectiveness of the proposed method. Compared with other video semantic segmentation methods, our framework achieves a better balance among false positive suppression, temporal stability, and reliable localization of occluded regions, which makes it more suitable for the occlusion segmentation task. These results verify that task-specific temporal fusion and boundary-aware optimization play an important role for video shot occlusion segmentation in real-world occlusion scenarios.

Acknowledgement

This work is supported in part by the Key Field Projects of Ordinary Universities in Guangdong Province (No. 2025ZDZX3050).

References

1. Chi, C., Zhang, S., Xing, J., Lei, Z., Li, S.Z., Zou, X.: Pedhunter: Occlusion robust pedestrian detector in crowded scenes. In: Proceedings of the AAAI conference on artificial intelligence. vol. 34, pp. 10639–10646 (2020)
2. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
3. Hesham, S.A.S., Liu, Y., Sun, G., Ding, H., Yang, J., Konukoglu, E., Geng, X., Jiang, X.: Exploiting temporal state space sharing for video semantic segmentation. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 24211–24221 (2025)
4. Hou, R., Ma, B., Chang, H., Gu, X., Shan, S., Chen, X.: Vrstc: Occlusion-free video person re-identification. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7183–7192 (2019)
5. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4700–4708 (2017)

6. Lazarow, J., Lee, K., Shi, K., Tu, Z.: Learning instance occlusion for panoptic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10720–10729 (2020)
7. Liao, J., Duan, H., Li, X., Xu, H., Yang, Y., Cai, W., Chen, Y., Chen, L.: Occlusion detection for automatic video editing. In: Proceedings of the 28th ACM international conference on multimedia. pp. 2255–2263 (2020)
8. Liao, J., Duan, H., Zhao, W., Yang, Y., Chen, L.: A light weight model for video shot occlusion detection. In: ICASSP 2022-2022 IEEE international conference on acoustics, speech and signal processing (ICASSP). pp. 3154–3158. IEEE (2022)
9. Liao, J., Duan, H., Zhao, W., Feng, K., Yang, Y., Chen, L.: A video shot occlusion detection algorithm based on the abnormal fluctuation of depth information. IEEE Transactions on Circuits and Systems for Video Technology **34**(3), 1627–1640 (2023)
10. Liu, Z., Ning, J., Cao, Y., Wei, Y., Zhang, Z., Lin, S., Hu, H.: Video swin transformer. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3202–3211 (2022)
11. Milletari, F., Navab, N., Ahmadi, S.A.: V-net: Fully convolutional neural networks for volumetric medical image segmentation. In: 2016 fourth international conference on 3D vision (3DV). pp. 565–571. Ieee (2016)
12. Nilsson, D., Sminchisescu, C.: Semantic video segmentation by gated recurrent flow propagation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 6819–6828 (2018)
13. Oh, S.W., Lee, J.Y., Xu, N., Kim, S.J.: Video object segmentation using space-time memory networks. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9226–9235 (2019)
14. Song, X., Zhao, K., Chu, W.S., Zhang, H., Guo, J.: Progressive refinement network for occluded pedestrian detection. In: European conference on computer vision. pp. 32–48. Springer (2020)
15. Sun, G., Liu, Y., Ding, H., Probst, T., Van Gool, L.: Coarse-to-fine feature mining for video semantic segmentation. In: proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3126–3137 (2022)
16. Sun, G., Liu, Y., Tang, H., Chhatkuli, A., Zhang, L., Van Gool, L.: Mining relations among cross-frame affinities for video semantic segmentation. In: European Conference on Computer Vision. pp. 522–539. Springer (2022)
17. Wei, Y., Tao, R., Wu, Z., Ma, Y., Zhang, L., Liu, X.: Occluded prohibited items detection: An x-ray security inspection benchmark and de-occlusion attention module. In: Proceedings of the 28th ACM international conference on multimedia. pp. 138–146 (2020)
18. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: Segformer: Simple and efficient design for semantic segmentation with transformers. Advances in neural information processing systems **34**, 12077–12090 (2021)
19. Zhang, X., Yan, Y., Xue, J.H., Hua, Y., Wang, H.: Semantic-aware occlusion-robust network for occluded person re-identification. IEEE Transactions on Circuits and Systems for Video Technology **31**(7), 2764–2778 (2020)
20. Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P.H., et al.: Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6881–6890 (2021)
21. Zhou, C., Yuan, J.: Occlusion pattern discovery for object detection and occlusion reasoning. IEEE Transactions on Circuits and Systems for Video Technology **30**(7), 2067–2080 (2019)